X-521-65-341

NASA TM X-55291

A COMPUTER PROGRAM TO IMPROVE STADAN STATION SCHEDULING

GPO PRICE $ ______

CFSTI PRICE(S) $ ______

Hard copy (HC) $2.00

Microfiche (MF) 50¢

ff 653 July 65

JULY 1965

GODDARD SPACE FLIGHT CENTER

GREENBELT, MARYLAND

X-521-65-341

A COMPUTER PROGRAM TO IMPROVE STADAN STATION SCHEDULING

by

Walter D. Davis
Goddard Space Flight Center
Greenbelt, Maryland

SUMMARY

34239

As a first step toward automatic STADAN station scheduling, a computer program is written to: (1) relieve the STADAN station scheduler from sifting through volumes of predicted satellite operations (i.e., Operational Predictions for Satellites) data to obtain the data necessary for station scheduling; (2) save the STADAN station scheduler the time, labor, and perhaps human errors of transferring the scheduling data to a station scheduling sheet or chart; and (3) place the scheduling data in a format more suitable for scheduling. The computer program is described in sufficient detail to enable anyone with a minimal understanding of general computer programming to operate it.

Author

CONTENTS

A COMPUTER PROGRAM TO IMPROVE STADAN STATION SCHEDULING*

INTRODUCTION

The Space Data Control Branch, in response to a request from the Network Engineering Operations Division, initiated a study to determine the applicability of high-speed digital computers to station scheduling. The study was to culminate with the Space Data Control Branch writing a computer program(s) to save the belabored station scheduler from being devoured by the ever increasing volumes and complexities of predicted satellite operations data which he utilizes to schedule the STADAN stations. The purpose of this technical note is to report that the Space Data Control Branch, in the person of the author, has written a computer program to: (1) edit the predicted satellite operations data, as calculated by the revised GSFC orbital programs, extracting only pertinent scheduling data; (2) generate or calculate new predicted satellite operations data when the time interval (during a station pass†) of successive predicted satellite operations data points (as calculated by the revised GSFC orbital programs) equals or exceeds two minutes; and (3) place the results of (1) and (2) in a format, as illustrated in Figure 1, which is more suitable for station scheduling. (NOTE: all satellite and STADAN station conflicts, at any arbitrary time, will be immediately apparent on "first sight" inspection).

During the initiation of the study, the Space Data Control Branch made plans to write, in addition to the aforementioned computer program (PHASE I), a second computer program (PHASE II) which would resolve all scheduling conflicts and otherwise make all station scheduling decisions which, heretofore, have been a manual operation. Those plans were later suspended. After PHASE I is put into operation and proven itself, an effort can be made to write PHASE II or a PHASE II-type computer program.

The PHASE I computer program is mostly written in the FORTRAN IV language. However, several control cards of the IBM 7090/7094 IBSYS OPERATING SYSTEM GENERALIZED SORTING SYSTEM are utilized along with several MAP routines for multireel output (i.e., the END OF FILE (EOF) mark is written automatically when the END OF TAPE (EOT) reflective spot is sensed

*STADAN is the denotation of SPACE TRACKING AND DATA ACQUISITION NETWORK.
STATION SCHEDULING is defined to be the allocation of the service time of a STADAN station to the satellites which pass through the station's field of tracking and data acquisition.
†STATION PASS is defined to be a passing of the satellite through the station's field of tracking and data acquisition.

FIGURE 1

TIME	640908 FTMYRS			252 JOBURG			LIMAPU			OOVERA			QUITOE			SNTAGO		
H M	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C
836			C		*				C						C			
837			C		*				C						7			C
838			C		*				C						C			C
839					*				C						C			C
840					*				6						C			C
841					*				C		*				C			C
842					*				C		*				C			C
843					*				C		*				C			C
844					*				C		*				C			C
845					*				C		*				C			C
846					*				C		*				C			C
847					*				*		*				*			5*
848					*				*		*							*
849					*				*		*					A		*
850					*				*		*					A		*
851					*						*					A		*
852					*						*					A		*
853					*						*					A		*
854					*						*					*		*

IDENTIFICATION

A-RELAY 1
B-RELAY 2
C-ECHO 2

while writing. Reel switching occurs after the EOF is written). The listed "deck names" (i.e., FRWB, FRWD, FSLI, FXEM, NOSYS, and GOSYS) within the PHASE I computer program, as shown in APPENDIX A, represent binary program decks which constitute modified IBM FORTRAN IV subroutines (written in MAP). These binary decks are utilized to enable T&DS' IBM 7094(s) at GSFC to: (1) read the next file, if desired by the programmer, after encountering the EOF mark on an input tape; and (2) use the record, as read the 100th time, if the "permanent read redundancy" error condition is encountered.

The FORTRAN statements, SORT control cards or instructions, MAP routines, and the aforementioned "deck names" of the PHASE I computer program are given in APPENDIX A.

SUBPROGRAMS OF PHASE I

The computer program of PHASE I consists of three subprograms as follows:

SUBPROGRAM 1: (1) Edit the predicted satellite operations data, as calculated by the revised GSFC orbital programs, extracting only pertinent scheduling data; (2) generate or calculate new predicted satellite operations data when the time interval (during a station pass) of successive predicted satellite operations data points (as calculated by the revised GSFC orbital programs) equals or exceeds two minutes; and (3) place the results of (1) and (2) in a format suitable for sorting internally on the IBM 7090/7094.

SUBPROGRAM 2: Sort the results of (3) of subprogram 1 on the IBM 7090/7094 with respect to date, time, station, and satellite.

SUBPROGRAM 3: Place the results of subprogram 2 in the final format as illustrated in Figure 1.

ON SUBPROGRAM 1

Input Data Cards

For a "run" of subprogram 1, (n + 1) + c data cards are input for n satellites of input data, where c is the total number of continuation cards. That is, excluding the continuation cards, one data card is input for each satellite of input data plus a data card which contains the total number (i.e., n) of satellites whose data are input into the computer for the "run" of subprogram 1.

The parameters on the input data cards of subprogram 1 are defined as follows:

TNSATL = total number of satellites whose data are input into the computer for a "run" of subprogram 1.

Z = an alphabetic or non-numerical character assigned to the satellite, whose data are input into the computer for a "run" of subprogram 1, to: (1) indicate, in the final output, that the satellite is in the field of tracking and data acquisition of a STADAN station but is not in the sunlight; and (2) identify the satellite and its column of data, in the final output, when TNSATL does not exceed the total number of alphabetic and "usable" special characters. (NOTE: If TNSATL exceeds the total number of alphabetic (26) and "usable" special characters, it is suggested that numbers or numerical characters, assigned to the satellites by subprogram 1, be used to identify the satellites and their columns of data in the final output (these numerical characters, obtainable from the on-line print-out of subprogram 1, would be inserted with the input data of subprogram 3 as the values of the parameter NOT(I)), with Z = A for all satellites whose data are input into the computer for the "run" of subprogram 1).

JPSI = An assigned elevation angle such that the subprogram 1 input data of any pass of the satellite through the field of tracking and data acquisition of any STADAN station will be deleted if the maximum elevation angle of the satellite during the pass does not equal or exceed it. (NOTE: With a minor modification in the subprogram 1 computer program, JPSI can be varied for each STADAN station which is scheduled to track and/or acquire data from the satellite.)

KEY = Number (0, 1, or 2) which enables the program operator to "inform" the computer of the number of files and their arrangement on the subprogram 1 input reel(s) of predicted operations data for a satellite:

KEY = 0 ⇒ every input reel of predicted operations data for the satellite has only one file.

KEY = 1 ⇒ first input reel of predicted operations data for the satellite has two and only two files while each of the other input reels, if any, has one and only one.

KEY = 2 ⇒ a "second" input reel of predicted operations data for the satellite has two and only two files while each of the other input reels including the first (i.e., reel which contains identifications and World Map data) has only one. (NOTE: Due to the format revision of the predicted satellite operations data and the subsequent modification of subprogram 1, it is not necessary to use "KEY = 2". When, due to identification and World Map data, two files appear on any input reel except the first, eliminate and don't input the reel(s) of identification and World Map data which precede the "two file reel." Then, treat the "two file reel" as the first input reel of predicted operations data for the satellite, and set K = 1. It is assumed, based on past operations, that the identification and World Map data will precede, on the input reels of predicted operations data for the satellite, the predicted operations data.)

NOS = total number of STADAN stations whose data are to be deleted during the "run" of subprogram 1.

NOSTAT = six character name of STADAN station whose data are to be deleted during the "run" of subprogram 1. (NOTE: the order of the names of the NOSTAT stations on the input data card(s) is not important).

NOSTAT(L) (L = 1,2,···, NOS) = NOSTAT in the Lth position of the linear array of NOSTAT(S) on the input data card.

It may be worthwhile to note that, in order to delete all of the predicted satellite operations data of a satellite, the program operator merely does <u>not</u> input the data of that satellite. The above statement assumes, based on past operations, that the predicted satellite operations data of two or more satellites are not contained on the same reel.

<u>Memory Allocations</u>

One hundred (100) core memory locations have been allocated for the STADAN stations (see the parameter JSTAT in the Dimension statement of subprogram 1, APPENDIX A). If necessary, the program operator may allocate additional (up to or beyond several thousand on the IBM 7094) core memory

locations for the STADAN stations. The core memory allocations of the parameter NOSTAT may be adjusted accordingly. Due to one or two adjustment "gimmicks" in the computer program, it will never be necessary to allocate additional core memory locations for any of the other parameters in the DIMENSION statement. In fact, the core memory allocations of JTIM, MINS, JEL, JCHAR, JDAY, JDATE, NTIM, NEL, NCHAR, NDAY, and NDATE, respectively, may be decreased if several FORTRAN statements are adjusted accordingly. (NOTE: the core memory allocations of each of the aforementioned parameters (i.e., JTIM, MINS, JEL, ...) may be decreased to (but not less than) two locations). However, if, within the predicted satellite operations data, there exists a large number of station passes such that the number of minutes in the duration of each station pass is greater than the core memory allocations of JTIM, MINS, JEL, JCHAR, JDAY, JDATE, NTIM, NEL, NCHAR, NDAY, and NDATE, respectively, the computer program (i.e., subprogram 1) would cause the computer to increase its "looping" and, hence, running time.

ON SUBPROGRAM 2

Information on the control cards of this subprogram may be obtained from any up-to-date manual on the IBM 7090/7094 IBSYS OPERATING SYSTEM GENERALIZED SORTING SYSTEM. For normal non-stop runs, the subprogram 2 computer program, as shown in APPENDIX A, may remain unchanged with the possible exception of the "REEL/n" field of the INPUT FILE control card. The number of reels of data input into the computer for a "run" of subprogram 2 is denoted by n in the "REELS/n" field. It is necessary to update the value of n for each "run" of subprogram 2. For example, suppose that the number of reels of subprogram 2 input data is 3 for a particular "run," then n is 3 and, hence, the INPUT FILE control card, for the particular "run," can be written as follows:

FILE, INPUT/1, REELS/3, MODE/D, DENSITY/H, BLOCKSIZE/5.

The "REELS/n" field may be omitted for a one-reel file (i.e., the number of reels of subprogram 2 input data is one).

ON SUBPROGRAM 3

Input Data Cards

For a "run" of subprogram 3, (n + m + 3) + c data cards, where c is the total number of continuation cards, are input for n satellites and m stations of input data. The parameters on these data cards are defined as follows:

DOTS 2 = two special characters which are used to separate, on the final print-out sheet, the data of a STADAN station from the data of the STADAN station(s) which is (are) adjacent to it. (NOTE: the character "•" followed by the blank character were used in the simulated "runs" and Figure 1).

DASH = two special characters which are used to indicate that the satellite or spacecraft is not in the field of tracking and data acquisition of the STADAN station. (NOTE: two blank characters were used in the simulated "runs" and Figure 1.)

TNSTAT = total number of distinct stations whose names are printed on-line during the computer "run" of subprogram 1.

TNSATL = same definition as TNSATL of subprogram 1. (NOTE: TNSATL of subprogram 3 is also equivalent to the total number of distinct satellites whose identifications are printed on-line during the computer "run" of subprogram 1, unless the program operator input, into the computer for a "run" of subprogram 1, the data of a satellite whose data are subsequently deleted because every STADAN station, through whose field of tracking and data acquisition the satellite is predicted to pass, is a NOSTAT station (see the definition of NOSTAT above.)

REELS = total number of reels of data which are output during the computer "run" of subprogram 2.

KSTAT = six (or less) character name of STADAN station. (EXAMPLE: BPOINT).

JK = station count or number assigned to KSTAT by the computer during the "run" of subprogram 1. (NOTE: all distinct KSTATs or names of STADAN stations whose data are input into the computer during a "run" of subprogram 1 are assigned numbers (by the computer) in ascending order, beginning with the number 1, according as they are input into the computer in a non-NOSTAT status (i.e., if KSTAT is a NOSTAT station for a satellite, it is neither assigned a number nor printed on-line during that computer "satellite loop." However, KSTAT could

be assigned a number and printed on-line during succeeding computer "satellite loops").)

NSATL = total number of distinct satellites whose identifications are printed on-line, during the computer "run" of subprogram 1, within the same record (i.e., the same line) as KSTAT. (NOTE: NSATL is also equivalent to the total number of distinct satellites which pass through the field of tracking and data acquisition of the KSTAT station and whose predicted data are input into the computer for the "run" of subprogram 1, unless the KSTAT station is a NOSTAT station to one or more of the distinct satellites.)

NUM = satellite count or number assigned to the satellite by the computer during the "run" of subprogram 1. (NOTE: all distinct satellites whose predicted data are input into the computer during a "run" of subprogram 1 are assigned numbers (by the computer) in ascending order, beginning with the number 1, according as their predicted data are input into the computer. If every STADAN station, through whose field of tracking and data acquisition the satellite is predicted to pass, is a NOSTAT station, the NUM of the satellite is not printed on-line even though a number is assigned to that satellite by the computer during the "run" of subprogram 1).

NUM (JK, I) (I = 1,2,···, NSATL) = NUM (i.e., satellite count) of the satellite (whose identification number is printed on-line, during the computer "run" of subprogram 1, within the same record as the KSTAT station whose station count is JK) which has the I th smallest NUM of all the satellites whose identification numbers are printed on-line, during the computer "run" of subprogram 1, within the same record as the KSTAT station whose station count is JK.*

NOT(I) = Z (see definition of Z above) of the satellite, whose NUM or satellite count is NUM(JK, I), followed by the blank character. (NOTE: NOT(I) is also printed on-line, during the computer "run" of subprogram 1, within the same

*Further explanation is given below.

record as the identification of the above satellite and, hence, NUM(JK,I)).

FDATE = the smallest (or earliest) START-DATE of all the START-DATES of the satellites whose predicted operations data are input into the computer for the "run" of subprogram 1.

FTIME = the smallest (or earliest) START-TIME of all the START-TIMES of the satellites whose predicted operations data are input into the computer for the "run" of subprogram 1.

LDATE = the largest (or oldest) STOP-DATE of all the STOP-DATES of the satellites whose predicted operations data are input into the computer for the "run" of subprogram 1.

LTIME = the largest (or oldest) START-TIME of all the START-TIMES of the satellites whose predicted operations data are input into the computer for the "run" of subprogram 1.

Z = Z of subprogram 1.

IDENT = identification number of the satellite. (NOTE: the identification number consists presently of five numerical characters).

JSATL = partial or full name of the satellite. (NOTE: JSATL = partial name of the satellite if the name of the satellite consists of seven or more characters. For example: if EXPLORER is the name of the satellite, JSATL = EXPLOR).

NOS = partial name of the satellite if the name of the satellite consists of seven or more characters. (Using the example from above, if EXPLORER is the name of the satellite, then NOS = ER followed by two blank characters.)

JSDAT = START-DATE of the satellite.

JSTIM = START-TIME of the satellite.

JXDAT = STOP-DATE of the satellite.

JXTIM = STOP-TIME of the satellite.

JPSI = JPSI of subprogram 1. (NOTE: in the format, exterior location 74, of subprogram 3 in APPENDIX A, space has been alloted to enable the program operator to insert a "decimal" character within the value of JPSI of subprogram 1.)

On the second input data card of subprogram 3, the values of the parameters TNSTAT, TNSATL, and REELS may be predetermined before the computer "runs" of subprograms 1, 2, and 3. The values of the parameters TNSTAT and TNSATL can also be easily determined from the subprogram 1 on-line print-out. However, as we indicated above, the value of TNSATL input into the computer for the "run" of subprogram 1 is different from the value of TNSATL determined from the subprogram 1 on-line print-out if the program operator input, into the computer for a "run" of subprogram 1, the data of a satellite whose data are subsequently deleted because every STADAN station, through whose field of tracking and data acquisition the satellite is predicted to pass, is a NOSTAT station. In this event, either value of TNSATL may be used (for the subprogram 3 input) since the predicted operational data of that satellite will not be processed or even considered during the computer "run" of subprogram 3. (NOTE: If the subprogram 1 input value of TNSATL is used, the set of input data cards whose parameters are Z, IDENT, JSATL, NOS, JSDAT, JSTIM, JXDAT, JXTIM, and JPSI should contain an input data card on every satellite whose predicted operational data are input for the computer "run" of subprogram 1. If the subprogram 1 on-line print-out value of TNSATL is used, the set of input data cards whose parameters are Z, IDENT, JSATL, NOS, JSDAT, JSTIM, JXDAT, JXTIM, and JPSI should contain an input data card on every satellite whose predicted operational data are input for the computer "run" of subprogram 1 except the satellites whose predicted operational data are deleted as described above.) In order to eliminate any possible variation of the value of TNSATL during the computer "runs" of subprograms 1, 2, and 3, the program operator should pre-check the station listing of each satellite, whose predicted operational data are input for the computer "run" of subprogram 1, and not input the data of any satellite whose predicted operational data would be deleted as described above.

On the set of input data cards whose parameters are JK, KSTAT, NSATL, NUM(JK, I), and NOT(I), I = 1,2,···, NSATL, the values of the aforementioned parameters should be obtained and determined from the subprogram 1 on-line print-out since the relationship of the values of JK, KSTAT, NUM(JK, I) and NOT(I), I = 1,2,···, NSATL, on each member (i.e., input data card) of the aforementioned set depends on the order in which the predicted operational data of the satellites are input into the computer for a "run" of subprogram 1.

Records of data consisting of the values of JK, KSTAT, NUM(JK, I), NOT(I), and IDENT, for some I, I = 1,2,···, NSATL, are printed on-line by the computer during the "run" of subprogram 1. An example of an on-line print-out of a subprogram 1 computer "run" is given as follows:

1	BPOINT	1	A	62681
2	FTMYRS	1	A	62681
1	BPOINT	2	B	72798
3	JOBURG	2	B	72798
2	FTMYRS	3	C	92671
3	JOBURG	4	D	42013

From the example above, it is easy to see that TNSTAT = 3 and TNSATL = 4. The set of input data cards whose parameters are JK, KSTAT, NSATL, NUM(JK, I), and NOT(I), I = 1,2,···, NSATL, consist of three members as follows:

FIRST INPUT CARD: JK = 1; KSTAT = BPOINT; NSATL = 2; NUM(1, 1) = 1; NOT(1) = A; NUM(1,2,) = 2; and NOT(2) = B.

SECOND INPUT CARD: JK = 2; KSTAT = FTMYRS; NSATL = 2; NUM(2,1) = 1; NOT(1) = A; NUM(2,2) = 3; and NOT(2) = C.

THIRD INPUT CARD: JK = 3; KSTAT = JOBURG; NSATL = 2; NUM(3,1) = 2; NOT(1) = B; NUM(3,2) = 4; and NOT(2) = D.

For a more automatic and perhaps faster operation from subprogram 1 through subprogram 3, subprogram 1 may be modified so that the array of input data whose parameters are TNSTAT, TNSATL, JK, KSTAT, NSATL, NUM(JK, I), and NOT(I), I = 1,2,···, NSATL, are either card punched on-line by the computer (NOTE: this operation (i.e., punching cards on-line) should be avoided on T&DS' IBM 7094 computers at GSFC) or written on a magnetic tape separate from the other output data tapes. The on-line punched cards or magnetic tape are/is later input into the computer for the "run" of subprogram 3. The subprogram 1 on-line print-out would no longer be vital to the operation of subprogram 3.

Memory Allocations

As in subprogram 1, one hundred (100) core memory locations have been allocated for the STADAN stations or, more specifically, for the values (i.e., names of the STADAN stations) of the parameter JSTAT. The core memory allocations for the parameters JSTAT and MSATL are and should always be the same (see the DIMENSION statement of subprogram 3). The core memory allocations for each of the parameters NUM, NOT, MOT, NSU, and JEL were allocated as a function of: (1) ten physical tape units (i.e., tape drives) for output; and/or (2) 150 character-wide print-out paper or sheets for the final results as illustrated in Figure 1.

Let: x be the number of core memory locations which are allocated for the names of the STADAN stations; y be the maximum number of tape drives, attached to the computer, which can be utilized for output during the "run" of subprogram 3; and Z be the total number of characters which can be written width-wise across the print-out paper which will be utilized for the final results. Then, the DIMENSION statement of subprogram 3 can be formulated, using FORTRAN language notations, as

$$\text{DIMENSION}\quad \text{NUM}\left(x, \frac{Z-6}{2}\right), \text{NOT}\left(y * \left(\frac{Z-6}{2}\right)\right), \text{MOT}\left(y * \left(\frac{Z-6}{2}\right)\right), \text{JSTAT}\,(x), \text{MSATL}\,(x),$$
$$\text{NSU}\,(y), \text{JEL}\left(\frac{Z-6}{2}\right)$$

(NOTE: Suprogram 3, Appendix A, is written to print the final results on 132 or more character-wide print-out paper. However, if the additional width on the "greater than 132" character-wide print-out paper is to be utilized, two FORTRAN statements must be adjusted according to the increase over 132.)

Format Updating

In Figure 1, the number of satellites serviced by a STADAN station is the same for all the stations. However, this is not always the case as the number of satellites serviced by two or more distinct STADAN stations may vary. Hence, in order to output the scheduling data in the final format, as illustrated in Figure 1, without "wasting" spaces on the print-out sheet between the data of two stations, it is necessary to update the output formats (i.e., exterior statement numbers 98, 99, 100, 101, 102, 103, 104, 105, 106, and 107) of subprogram 3.

The formats whose exterior statement numbers are 98, 99, 100, 101, 102, 103, 104, 105, 106, and 107 are generalized and formulated in APPENDIX B where:

MSATL(I), I = 1,2,···, NSU(1), NSU(1) + 1,···, NSU(10), is the number of satellites serviced by the STADAN station whose station count (i.e., JK) is I.

NSU(1) is the number of STADAN stations whose data are output on logical unit A3.

NSU(2) is the number of STADAN stations whose data are output on logical unit B5.

....

NSU(10) is the number of STADAN stations whose data are output on logical unit A8.

If MSATL(I), I = 1,2,···, NSU(10), is 2, it is suggested that the name (i.e., KSTAT) of the STADAN station whose station count is I be input, on the input data cards of subprogram 3, as a four-character name, instead of a six-character name, followed by two blank characters. Otherwise, the six-character names of at least two STADAN stations will be written together in the heading of the final print-out with no space between them.

If MSATL(I), I = 1,2,···, NSU(10), is 1, the formulated formats of APPENDIX B do not hold true. In order to rectify this, it is suggested that: (1) the name (i.e., KSTAT) of the STADAN station whose station count is I be input, on the input data cards of subprogram 3, as a two character name followed by four blank characters; (2) the "A6" (of the aforementioned formats) which precede the expression which contains MSATL(I) be changed to "A2"; and (3) set (2 * MSATL(I) - 4), the expression which contains MSATL(I), equal to 2.

Elimination of Computer Hang-Up and Unneccessary Dumps

If a logical tape unit which is referenced in subprogram 3 is not used during a "run" of subprogram 3, the 7094 computer system of T&DS at GSFC will still select and unload that logical tape unit after the "run". This may often result into a computer hang-up and an unnecessary dump of core storage. To avoid this, appropriate $NAME card(s) should be inserted into the subprogram 3 deck after the $IBJOB card, along with the proper instruction on the instruction card. For example, in a short simulated "run" of subprogram 3, only logical units A5 and A3 were utilized during the computer "run". In order to stop the computer from selecting and attempting to unload logical units A6, B5, B6, C1, C2, A7, B7, B8, B9, and A8 all of subprogram 3, nine $NAME cards were inserted into the

subprogram 3 deck after the $IBJOB card in the form

```
COLUMN 1              16
       |              |
       $NAME          .UNI . = .UN19.
```

Where I = 06, 15, 16, 21, 22, 07, 17, 18, 08. Then, an output reel for logical unit B9 (i.e., .UN19.) was requested on the instruction card.

The MAP routine for multireel output on a particular logical tape unit should be inserted into the subprogram 3 deck for a computer "run" if and only if that logical unit will be utilized during that "run" (or all MAP routines for multireel output on the logical tape units which will not be utilized during a computer "run" should be removed from the subprogram 3 computer program of APPENDIX A for that "run"). If a MAP routine for multireel output on a logical tape unit is inserted into the subprogram 3 deck for a computer "run", the computer will select and attempt to unload the logical tape unit after that "run".

<u>Output</u>

If the data at some time t, as illustrated in Figure 1, exceed 132 characters, two or more physical tape units (i.e., tape drives) are used to output the data in the format as illustrated in Figure 1. If the data at some time t, as illustrated in Figure 1, exceed 1320 characters (i.e., 132 characters times 10 tape drives) or if more than ten tape drives are needed for output, the computer will stop its execution of subprogram 3, after printing an "informative" message on-line, and return control to the Basic Monitor. (IMPORTANT NOTE: If ten tape drives are used for output during the computer "run" of subprogram 3, logical tape unit A2 (i.e., SYSIN1, the system input tape unit for the T&DS' IBM 7094 computers at GSFC) is eliminated by subprogram 3, and the computer "hangs-up" <u>after</u> the subprogram 3 computer "run".)

The output tapes of subprogram 3 are printed off-line, by the IBM 1401 computer, on 132 character-wide paper or sheets. The print-out sheets of the output tape(s) of each tape drive used for output during the subprogram 3 computer "run" are bursted and then bound into a book. Within each book of scheduling data, there exist no unused "character space" between the data of two adjacent STADAN stations even though the number of satellites serviced by one of the stations is not the same as the number of satellites serviced by the other. However, an unused "character space(s)" may exist at the end, width-wise, of each page or sheet of a book of scheduling data since all the scheduling data of each STADAN station are contained within one book of scheduling data. (NOTE: subprogram 3 can be modified or revised so that unused "character spaces" can not

exist at the end, width-wise, of each page of a book of scheduling data when the station count (i.e., JK) of each STADAN station within the book is less than the station count of any STADAN station within another book of the same "run." The result of the above revision would be that the data of some STADAN stations would be "divided" or "broken-up" and placed into two separate books with a part of the data in one book and the other part in a second book. This appeared to be bad human engineering from the reader or STADAN station scheduler's viewpoint).

PASS-CHAINS OF FINAL OUTPUT

In the final print-out or books of scheduling data, it is easy to recognize the station passes of each satellite by observing the vertical "chains" of data, at one minute intervals, under the satellite's notation or identification character(s) as illustrated in Figure 1. Henceforth, a vertical chain of data at one minute intervals will be called a pass-chain.

Regardless of the variations of the time increments of successive data points of predicted satellite operations, no break or breaks will ever occur in a pass-chain except when, during a station pass of more than 1 day, the day count increment (disregarding the 365 or 366 days/year) of two successive data points of predicted satellite operations exceeds 1 day. Scheduling data are not generated, by subprogram 1, for the date or dates on which no data point of predicted satellite operations is computed. If at least 1 data point, of predicted satellite operations for a station pass of any duration, is computed for a date, additional scheduling data will be generated, by means of subprogram 1, to complete the pass-chain for that date. To guard against breaks in the pass-chain, the time increment of any two successive data points, of predicted satellite operations for a station pass, should never exceed 24 hours or 1440 minutes. (NOTE: While it is recognized that the above case is remote or far-fetched, the situation could be encountered with the Syncom-type satellites and deep space probe vehicles.)

METHODS FOR HANDLING EXTRA LARGE NUMBERS OF STATIONS AND SATELLITES

Each of the T&DS IBM 7094 computers (i.e., A, B, and C) at GSFC has a maximum of 14 physical tape units (i.e., tape drives) even though each has 30 logical tape units. Subprogram 3 can use up to 12 of the 14 physical tape units, 2 for input and 10 for output. The 7094 computer system will not allow subprogram 3 to use the remaining 2 physical tape units.

The 10 output tape units, as used in subprogram 3 with 132 character-wide print-out sheets, are presently adequate for all normal operations. However, for an abnormal operation (i.e., an extra large number of stations) and with the number of satellites in orbit steadily increasing, the 10 output tape units, as used in subprogram 3 with 132 character-wide print-out sheets, may/may soon be inadequate. In order to make the 10 output tape units, as used in subprogram 3 with 132 character-wide print-out sheets, adequate for all present and future operations including extra large numbers of stations and satellites, it is necessary to devise methods or techniques to overcome the above limitations. Three such methods are listed as follows:

METHOD 1: The names of the STADAN stations which are listed in the predicted satellite operations data (i.e., the input of subprogram 1), and are not NOSTAT stations (see definition of NOSTAT, above) are separated into two or more sets, say set A, set B, etc. . . . Then:

(a) the members of all of the above sets except, say, the set A are punched on subprogram 1 input data cards as NOSTAT stations. Now, the above input data cards, other appropriate data cards, and <u>all</u> of the predicted satellite operations data are input, in the usual manner, into the computer for a "run" of subprogram 1. Using the output of subprogram 1, subprograms 2 and, finally, 3 are "run" in the usual manner, also. The final result of these "runs" (i.e., subprograms 1, 2, and 3) is a set of books of scheduling data whose STADAN stations are members of set A.

(b) the members of all the sets (i.e., set A, set B, etc. . .) except set B are punched on subprogram 1 input data cards as NOSTAT stations. Then, these input data cards, other appropriate data cards and, again, all of the predicted satellite operations data are input, in the usual manner, into the computer for a "run" of subprogram 1. Using the output of subprogram 1, subprogram 2 and, finally, 3 are "run" in the usual manner. The final result is a set of books of scheduling data whose STADAN stations are members of set B.

(c) etc. . . . .

Of course, this method increases the computer time. However, the placement of the scheduling data within the books is virtually unaffected by the method.

METHOD 2: the names of the satellites, each of which is predicted to pass through the field of tracking and data acquisition of at least one non-NOSTAT station and whose predicted operations data are to be input into the computer for the "run" of subprogram 1, are separated into two or more sets, say set G, set H, etc. . . . Then:

(a) the predicted operations data of the satellites whose names are members of one and only one of the above sets, say set G, are input into the computer for a "run" of subprogram 1. The input data cards of subprogram 1 are appropriately adjusted for the "run." Using the output of subprogram 1, subprograms 2 and, finally, 3 are "run" in the usual manner. The final result of these "runs" (i.e., subprograms 1, 2, and 3) is a set of books of scheduling data whose satellites are members of set G.

(b) the predicted operations data of the satellites whose names are members of set H are input into the computer for another "run" of subprogram 1. The input data cards of subprogram 1 are appropriately adjusted, again. Using the output of subprogram 1, subprograms 2 and, finally, 3 are then "run" in the usual manner. The final result is a set of books of scheduling data whose satellites are members of set H.

(c) etc. . . .

The computer executing time, excluding compiler time, is not significantly increased by this method. However, the placement of the scheduling data within the book is affected.

METHOD 3: An appropriate combination of methods 1 and 2, above.

EXPLANATION OF FIGURE 1

A blank at time t in column A, B, or C under a particular station indicates that RELAY 1, RELAY 2, or ECHO 2 is not in the field of tracking and data acquisition of that station at time t. For example, at time 0836, RELAY 1 and RELAY 2 are not in the field of tracking and data acquisition of FTMYRS.

A, B, or C at time t in column A, B, or C under a particular station indicates that RELAY 1, RELAY 2, or ECHO 2 is in the field of tracking and data acquisition of that station but is not in sunlight at time t. For example, at time 0836, ECHO 2 is in the field of tracking and data acquisition of FTMYRS, LIMAPU, and QUITOE but is not in the sunlight.

An "*" at time t in column A, B, or C under a particular station indicates that RELAY 1, RELAY 2, or ECHO 2 is in the field of tracking and data acquisition of that station and is in sunlight at time t. For example, at time 0836, RELAY 2 is in the field of tracking and data acquisition of JOBURG and is in the sunlight.

At time 0837, 7 in column C under station QUITOE indicates that ECHO 2 is at maximum elevation angle in the field of tracking and data acquisition of QUITOE but is not in sunlight, and the maximum elevation angle ranges from 70-79 degrees.

At time 0847, 5* in column C under station SNTAGO indicates that ECHO 2 is at maximum elevation angle in the field of tracking and data acquisition of SNTAGO and is in sunlight, and the maximum elevation angle ranges from 50-59 degrees.

The character "●" followed by the blank character were used to separate the data of the stations (see the parameter DOTS2 of the INPUT DATA CARDS of SUBPROGRAM 3, above).

CONCLUSION

The computer program is capable of: (1) deleting, from the predicted satellite operations data (as calculated by the revised GSFC orbital programs), the data of any orbital parameter, station pass, STADAN station, and/or satellite; (2) "handling" or processing the predicted operations data of any satellite, made or unmade, including Syncom or a Syncom-type satellite (which would have a single continuous pass from START-DATE to STOP-DATE); (3) imposing no limits on the duration of any pass (that is to say, the computer program can "handle" a station pass of any duration); (4) generating or calculating new predicted satellite operations data when the time interval (during a station pass) of successive predicted satellite operations data points equals or exceeds two minutes; and (5) processing, by methods 1, 2, and 3 as described above, the predicted operations data of extra large numbers of satellites and stations.

In addition to improving STADAN station scheduling as described above, the computer program may also serve as a "stepping stone" and catalyst to a Phase II-type computer program for more fully automated station scheduling.

The utilization of the computer program (and perhaps later a Phase II-type computer program) would result in a significant saving in manpower as well as a faster and more efficient operation.

BLANK PAGE

APPENDIX A

SUBPR@GRAM 1

```
$IBSYS
$ID R02T        WDD521A, X4189
$*                DEAR @PERAT@R, PLEASE M@UNT THE TAPES,AS DESIGNATED @N
$*                INSTRUCTI@N CARD,@N A5,A6,AND B5. A3 AND B6 ARE MY @UTPUT
$*                UNITS. SAVE MY B6 @UTPUT TAPES.
$*                RESTART MACHINE. THE MACHINE WILL READ AND REWIND A5,A6,
$*                AND B5,SEQUENTIALLY AND CYCLICALLY,UNTIL ALL TAPES ARE
$*                READ. AFTER A5,A6,@R B5 IS REW@UND, M@UNT,IMMEDIATELY,THE
$*                NEXT TAPE AS SH@WN @N INSTRUCTI@N CARD.
$PAUSE
$DATE             052065
$EXECUTE          IBJ@B
$IBJ@B            G@
$IBFTC MAIN
C     STATI@N SCHEDULING,SUBPR@GRAM 1 @F PHASE 1
      L@GICAL E@F
      C@MM@N/SYSE@F/E@F
      E@F=.FALSE.
      CALL SYSG@(34)
      CALL SYSG@(35)
      INTEGER X,@P,EN,STA,TIME,Z,TNSATL
     0DATA @P/6H@P    /,EN/6HEN    /,STA/6H(STATI/,TIME/6H(TIME /,ICHAR/
     16H*     /
     0DIMENSI@N N@STAT(99),JSTAT(100),KTIM(4),JTIM(900),MINS(900),JEL(90
     10),JCHAR(900),JDAY(900),JDATE(900),NTIM(900),NEL(900),NCHAR(900),N
     2DAY(900),NDATE(900)
      READ (2,3)TNSATL
    3 F@RMAT (I3)
      K=0
      J=1
      D@ 4 I=1,TNSATL
      READ (2,5)Z,JPSI,KEY,N@S,(N@STAT(L),L=1,N@S)
    5 F@RMAT (A1,A4,I1,I2,99A6)
      IF (KEY)16,17,16
   16 G@T@ (18,19,20),J
   18 READ (5,21)KSKIP
      IF (E@F)G@T@ 22
      G@T@ 18
   19 READ (6,21)KSKIP
      IF (E@F)G@T@ 22
      G@T@ 19
   20 READ (15,21)KSKIP
      IF (E@F)G@T@ 22
      G@T@ 20
   21 F@RMAT (I1)
   22 E@F=.FALSE.
      IF (KEY-1)232,17,23
   23 KEY=1
      G@T@ (24,25,26),J
   24 REWIND 5
      J=2
      G@T@ 19
   25 REWIND 6
      J=3
      G@T@ 20
   26 REWIND 15
      J=1
      G@T@ 18
```

```
17 GOTO (27,28,29),J
27 READ (5,30)IDENT,X
   IF (EOF)GOTO 31
   IF (X-STA)32,33,32
32 IF (X-TIME)27,34,27
28 READ (6,30)IDENT,X
   IF (EOF)GOTO 31
   IF (X-STA)35,33,35
35 IF (X-TIME)28,34,28
29 READ (15,30)IDENT,X
   IF (EOF)GOTO 31
   IF (X-STA)36,33,36
36 IF (X-TIME)29,34,29
30 FORMAT (29X,A5,1X,A6)
34 PRINT 37,IDENT
370FORMAT (1H0,50HSTOPPING----PREDIXS ARE TIME SORTED FOR SATELLITE ,
  1A5,35H . USE ONLY STATION SORTED PREDIXS.)
   GOTO 232
31 EOF=.FALSE.
   IF (KEY-1)38,17,38
38 GOTO (39,40,41),J
39 REWIND 5
   J=2
   GOTO 28
40 REWIND 6
   J=3
   GOTO 29
41 REWIND 15
   J=1
   GOTO 27
33 GOTO (42,43,44),J
42 READ (5,21)KSKIP
   IF (EOF)GOTO 45
   GOTO 46
43 READ (6,21)KSKIP
   IF (EOF)GOTO 45
   GOTO 46
44 READ (15,21)KSKIP
   IF (EOF)GOTO 45
   GOTO 46
45 EOF=.FALSE.
   GOTO (47,48,49),J
47 REWIND 5
   J=2
   GOTO 43
48 REWIND 6
   J=3
   GOTO 44
49 REWIND 15
   J=1
   GOTO 42
46 GOTO (50,51,52),J
50 READ (5,53)KSTC,KSTAT,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
   IF (EOF)GOTO 54
   GOTO 55
51 READ (6,53)KSTC,KSTAT,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
   IF (EOF)GOTO 54
   GOTO 55
52 READ (15,53)KSTC,KSTAT,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
   IF (EOF)GOTO 54
```

```
      GØTØ 55
   53 FØRMAT (1X,A2,1X,A6,A1,6X,A3,1X,A6,1X,4A1,14X,A4,16X,A1)
   54 EØF=.FALSF.
      GØTØ (56,57,58),J
   56 REWIND 5
      J=2
      GØTØ 51
   57 RFWIND 6
      J=3
      GØTØ 52
   58 REWIND 15
      J=1
      GØTØ 50
   55 DØ 59 M=1,NØS
      IF (KSTAT-NØSTAT(M))59,60,59
   59 CØNTINUF
      JSTC=KSTC
      IF (K)61,62,61
   61 DØ 63 JK=1,K
      IF (JSTAT(JK)-KSTAT)63,64,63
   63 CØNTINUF
   62 K=K+1
      JK=K
      JSTAT(JK)=KSTAT
   64 NBIG=0
  196 NPASS=0
   65 DØ 66 N=1,900
      DØ 67 M=1,4
      KTIM(M)=KTIM(M)/(2**30)
   67 CØNTINUF
      JTIM(N)=KTIM(1)*(10**3)+KTIM(2)*(10**2)+KTIM(3)*10+KTIM(4)
      MINS(N)=KTIM(3)*10+KTIM(4)
      JFL(N)=KFL
      JCHAR(N)=KCHAR
      JDAY(N)=KDAY
      JDATE(N)=KDATE
   80 GØTØ (68,69,70),J
   68 READ (5,153)KSTC,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
      IF (FØF)GØTØ 71
      GØTØ 72
   69 READ (6,153)KSTC,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
      IF (EØF)GØTØ 71
      GØTØ 72
   70 READ (15,153)KSTC,KPASS,KDAY,KDATE,KTIM,KEL,KCHAR
      IF (FØF)GØTØ 71
      GØTØ 72
  153 FØRMAT (1X,A2,7X,A1,6X,A3,1X,A6,1X,4A1,14X,A4,16X,A1)
   71 EØF=.FALSF.
      GØTØ (73,74,75),J
   73 REWIND 5
      J=2
      GØTØ 69
   74 REWIND 6
      J=3
      GØTØ 70
   75 REWIND 15
      J=1
      GØTØ 68
   72 IF (JSTC-KSTC)77,76,77
   76 IF (KPASS-ICHAR)66,78,66
```

```
77 GOTO (81,82,83),J
81 READ (5,21)KSKIP
   IF (EOF)GOTO 84
   GOTO 85
82 READ (6,21)KSKIP
   IF (EOF)GOTO 84
   GOTO 85
83 READ (15,21)KSKIP
   IF (EOF)GOTO 84
   GOTO 85
84 EOF=.FALSE.
   GOTO (86,87,88),J
86 REWIND 5
   J=2
   GOTO 82
87 REWIND 6
   J=3
   GOTO 83
88 REWIND 15
   J=1
   GOTO 81
85 IF (KSTC-OP)79,80,79
66 CONTINUE
   NPASS=900
   N=900
79 GOTO (89,90,91),J
89 READ (5,92)KSTC
   IF (EOF)GOTO 93
   GOTO 78
90 READ (6,92)KSTC
   IF (EOF)GOTO 93
   GOTO 78
91 READ (15,92)KSTC
   IF (EOF)GOTO 93
   GOTO 78
92 FORMAT (1X,A2)
93 EOF=.FALSE.
   GOTO (94,95,96),J
94 REWIND 5
   J=2
   GOTO 90
95 REWIND 6
   J=3
   GOTO 91
96 REWIND 15
   J=1
   GOTO 89
78 IF (NBIG)97,98,97
98 JELM=0
97 DO 99 M=1,N
   JELM=MAX0(JEL(M),JELM)
99 CONTINUE
   IF (JELM-JPSI)100,101,101
101 IN=1
   MX=1
   IX=1
   J630=2
   J624=2
   J612=2
   J632=2
```

```
      J130=2
      J121=2
      J132=2
      J139=2
      IF (NBIG)102,645,102
  645 IF (N-1)601,634,601
  601 IN=IN+1
  103 IF (JDATE(IN)-JDATE(MX))106,625,106
  625 IF (MINS(MX)-000059)611,622,611
  622 IF (JTIM(IN)-JTIM(MX)-000041)603,604,623
  623 NTIM(IX)=JTIM(MX)
      NEL(IX)=JEL(MX)
      NCHAR(IX)=JCHAR(MX)
      NDAY(IX)=JDAY(MX)
      NDATE(IX)=JDATE(MX)
      IF (IX-900)624,613,613
  624 IF (JEL(MX)-JELM)627,628,628
  628 JEL(MX)=JEL(IN)
  627 IX=IX+1
      JTIM(MX)=JTIM(MX)+000041
      MINS(MX)=0
      IF (JDATE(IN)-JDATE(MX))605,611,605
  611 IF (JTIM(IN)-JTIM(MX)-000001)603,604,605
  603 IF (JEL(IN)-JELM)606,607,607
  607 JEL(MX)=JEL(IN)
      JCHAR(MX)=JCHAR(IN)
  606 IF (IN-N)601,608,601
  604 NTIM(IX)=JTIM(MX)
      NEL(IX)=JEL(MX)
      NCHAR(IX)=JCHAR(MX)
      NDAY(IX)=JDAY(MX)
      NDATE(IX)=JDATE(MX)
      MX=IN
      IF (IX-900)630,609,609
  630 IX=IX+1
      IF (IN-N)601,608,601
  605 NTIM(IX)=JTIM(MX)
      NEL(IX)=JEL(MX)
      NCHAR(IX)=JCHAR(MX)
      NDAY(IX)=JDAY(MX)
      NDATE(IX)=JDATE(MX)
      IF (IX-900)612,631,631
  612 IF (JEL(MX)-JELM)629,626,626
  626 JEL(MX)=JEL(IN)
  629 IX=IX+1
      JTIM(MX)=JTIM(MX)+000001
      MINS(MX)=MINS(MX)+000001
      IF (JDATE(IN)-JDATE(MX))107,625,107
  107 IF (MINS(MX)-000059)605,106,106
  609 J630=0
      GOTO 108
  613 J624=0
      GOTO 108
  631 J612=0
      GOTO 108
  608 IF (NPASS)640,634,640
  634 NTIM(IX)=JTIM(MX)
      NEL(IX)=JEL(MX)
      NCHAR(IX)=JCHAR(MX)
      NDAY(IX)=JDAY(MX)
```

```
      NDATE(IX)=JDATE(MX)
      GOTO 108
  640 NJTIM=JTIM(MX)
      NJFL=JFL(MX)
      NJCHAR=JCHAR(MX)
      NJDAY=JDAY(MX)
      NJDATE=JDATE(MX)
      NMINS=MINS(MX)
      IX=IX-1
      GOTO 108
  106 IF (JTIM(MX)-002359)110,111,111
  110 IF (MINS(MX)-000059)605,623,605
  111 NTIM(IX)=JTIM(MX)
      NFL(IX)=JFL(MX)
      NCHAR(IX)=JCHAR(MX)
      NDAY(IX)=JDAY(MX)
      NDATE(IX)=JDATE(MX)
      IF (IX-900)632,633,633
  632 IF (JEL(MX)-JELM)112,113,113
  113 JEL(MX)=JFL(IN)
  112 IX=IX+1
      JTIM(MX)=0
      MINS(MX)=0
      JDAY(MX)=JDAY(IN)
      JDATE(MX)=JDATE(IN)
      GOTO 611
  633 J632=0
      GOTO 108
  102 IF (JDATE(IN)-NJDATE)114,115,114
  115 IF (NMINS-000059)116,117,116
  117 IF (JTIM(IN)-NJTIM-000041)118,119,120
  120 NTIM(IX)=NJTIM
      NFL(IX)=NJFL
      NCHAR(IX)=NJCHAR
      NDAY(IX)=NJDAY
      NDATE(IX)=NJDATE
      IF (IX-900)121,122,122
  121 IF (NJFL-JFLM)123,124,124
  124 NJFL=JFL(IN)
  123 IX=IX+1
      NJTIM=NJTIM+000041
      NMINS=0
      IF (JDATE(IN)-NJDATE)125,116,125
  116 IF (JTIM(IN)-NJTIM-000001)118,119,125
  118 IF (JEL(IN)-JELM)126,127,127
  127 NJEL=JEL(IN)
      NJCHAR=JCHAR(IN)
  126 IF (IN-N)128,129,128
  129 IF (NPASS)196,301,196
  301 NTIM(IX)=NJTIM
      NFL(IX)=NJFL
      NCHAR(IX)=NJCHAR
      NDAY(IX)=NJDAY
      NDATE(IX)=NJDATE
      GOTO 108
  128 IN=IN+1
      GOTO 102
  119 NTIM(IX)=NJTIM
      NFL(IX)=NJEL
      NCHAR(IX)=NJCHAR
```

```
      NDAY(IX)=NJDAY
      NDATE(IX)=NJDATE
      MX=IN
      IF (IX-900)130,131,131
  130 IX=IX+1
      IF (IN-N)601,608,601
  125 NTIM(IX)=NJTIM
      NEL(IX)=NJFL
      NCHAR(IX)=NJCHAR
      NDAY(IX)=NJDAY
      NDATE(IX)=NJDATE
      IF (IX-900)132,133,133
  132 IF (NJFL-JFLM)134,135,135
  135 NJFL=JFL(IN)
  134 IX=IX+1
      NJTIM=NJTIM+000001
      NMINS=NMINS+000001
      IF (JDATE(IN)-NJDATE)136,115,136
  136 IF (NMINS-000059)125,114,114
  131 J130=0
      GOTO 108
  122 J121=0
      GOTO 108
  133 J132=0
      GOTO 108
  114 IF (NJTIM-002359)137,138,138
  137 IF (NMINS-000059)125,120,125
  138 NTIM(IX)=NJTIM
      NEL(IX)=NJFL
      NCHAR(IX)=NJCHAR
      NDAY(IX)=NJDAY
      NDATE(IX)=NJDATE
      IF (IX-900)139,140,140
  139 IF (NJFL-JFLM)141,142,142
  142 NJEL=JFL(IN)
  141 IX=IX+1
      NJTIM=0
      NMINS=0
      NJDAY=JDAY(IN)
      NJDATF=JDATF(IN)
      GOTO 116
  140 J139=0
  108 DO 167 M=1,IX
      IF (NEL(M)-JFLM)168,169,169
  169 IF (NCHAR(M)-ICHAR)170,171,170
  171 WRITE (16,172)NDATE(M),NTIM(M),JK,I,NDAY(M),NEL(M),NCHAR(M)
  172 FORMAT (2H0C,1X,A6,1X,I4,1X,I3,1X,I3,1X,A3,2X,A1,A1)
      GOTO 167
  168 IF (NCHAR(M)-ICHAR)173,174,173
  174 NFL(M)=NCHAR(M)
      GOTO 170
  173 NFL(M)=Z
  170 WRITE (16,175)NDATE(M),NTIM(M),JK,I,NDAY(M),NEL(M)
  175 FORMAT (2H0C,1X,A6,1X,I4,1X,I3,1X,I3,1X,A3,2X,A1,1H )
  167 CONTINUE
      IF (IX-900)176,177,176
  177 IF (J630)178,179,178
  178 IF (J624)180,181,180
  180 IF (J612)182,183,182
  182 IF (J632)184,185,184
```

```
184 IF (J130)186,187,186
186 IF (J121)188,189,188
188 IF (J132)190,191,190
190 IF (J139)176,193,176
179 J630=2
    IX=0
    GØTØ 630
181 J624=2
    IX=0
    GØTØ 624
183 J612=2
    IX=0
    GØTØ 612
185 J632=2
    IX=0
    GØTØ 632
187 J130=2
    IX=0
    GØTØ 130
189 J121=2
    IX=0
    GØTØ 121
191 J132=2
    IX=0
    GØTØ 132
193 J139=2
    IX=0
    GØTØ 139
176 IF (NPASS)194,100,194
194 NBIG=4
    GØTØ 196
100 IF (JSTC-KSTC)197,64,197
197 PRINT 199,JK,JSTAT(JK),I,Z,IDENT
199 FØRMAT (7X,I3,3X,A6,8X,I3,3X,A1,3X,A5)
223 IF (KSTC-FN)46,227,46
 60 JSTC=KSTC
218 GØTØ (200,201,202),J
200 READ (5,92)KSTC
    IF (EØF)GØTØ 204
    IF (JSTC-KSTC)205,200,205
201 READ (6,92)KSTC
    IF (EØF)GØTØ 204
    IF (JSTC-KSTC)205,201,205
202 READ (15,92)KSTC
    IF (EØF)GØTØ 204
    IF (JSTC-KSTC)205,202,205
204 EØF=.FALSE.
    GØTØ (206,207,208),J
206 REWIND 5
    J=2
    GØTØ 201
207 REWIND 6
    J=3
    GØTØ 202
208 REWIND 15
    J=1
    GØTØ 200
205 GØTØ (209,210,211),J
209 READ (5,21)KSKIP
    IF (EØF)GØTØ 212
    GØTØ 213
```

```
  210 READ (6,21)KSKIP
      IF (EØF)GØTØ 212
      GØTØ 213
  211 READ (15,21)KSKIP
      IF (EØF)GØTØ 212
      GØTØ 213
  212 EØF=.FALSE.
      GØTØ (214,215,216),J
  214 REWIND 5
      J=2
      GØTØ 210
  215 REWIND 6
      J=3
      GØTØ 211
  216 REWIND 15
      J=1
      GØTØ 209
  213 IF (KSTC-ØP)217,218,217
  217 GØTØ (219,220,221),J
  219 READ (5,92)KSTC
      IF (EØF)GØTØ 222
      GØTØ 223
  220 READ (6,92)KSTC
      IF (EØF)GØTØ 222
      GØTØ 223
  221 READ (15,92)KSTC
      IF (EØF)GØTØ 222
      GØTØ 223
  222 EØF=.FALSE.
      GØTØ (224,225,226),J
  224 REWIND 5
      J=2
      GØTØ 220
  225 REWIND 6
      J=3
      GØTØ 221
  226 REWIND 15
      J=1
      GØTØ 219
  227 GØTØ (228,229,230),J
  228 REWIND 5
      J=2
      GØTØ 4
  229 REWIND 6
      J=3
      GØTØ 4
  230 REWIND 15
      J=1
    4 CØNTINUE
      PRINT 231
  231 FØRMAT (1H0,10HEND ØF RUN)
      END FILE 16
  232 STØP
      END
$IBMAP UN16
       ENTRY    .UN16.
.UN16. PZE      UNIT16
UNIT16 FILE     ,B(2),READY,INØUT,BLK=22,MULTIREEL,BCD,HØLD,NØLIST
       END
```

```
$IBLDR FRWB
(RELØCATABLE BINARY DECK FØR SUBRØUTINE FRWB)

$IBLDR FRWD
(RELØCATABLE BINARY DECK FØR SUBRØUTINE FRWD)

$IBLDR FSLI
(RELØCATABLE BINARY DECK FØR SUBRØUTINE FSLI)

$IBLDR FXEM
(RELØCATABLE BINARY DECK FØR SUBRØUTINE FXEM)

$IBLDR NØSYS
(RELØCATABLE BINARY DECK FØR SUBRØUTINE NØSYS)

$IBLDR GØSYS
(RELØCATABLE BINARY DECK FØR SUBRØUTINE GØSYS)

(END-ØF-FILE CARD)

(DATA CARDS)

(END-ØF-FILE CARD)
$EMSYS
(END-ØF-FILE CARD)
```

SUBPRØGRAM 2

```
$IBSYS
$ID R02T       WDD521A, X4189
$RELEASE         SYSPP1
$RELEASE         SYSUT1
$RELEASE         SYSUT2
$RELEASE         SYSUT3
$RELEASE         SYSUT4
$ATTACH          RDA
$AS              SYSIN1
$EXECUTE         SØRT
       ØPTIØN,MAP,HISTØRY,CARDS,TAPES,NØCKPT
       CHANNEL,INPUT/B,MERGE/(A,B)
       FILE,INPUT/1,REELS/1,MØDE/D,DENSITY/H,BLØCKSIZE/5
       FILE,ØUTPUT,MØDE/D,DENSITY/H,BLØCKSIZE/5
       RECØRD,TYPE/F,LENGTH/5,FIELD/(3,6,1,4,1,3,1,3)
       SØRT,SEQUENCE/C,ØRDER/3,FIELD/(2,4,6,8),FILE/1
       END
$IBSYS
$STØP
```

SUBPRØGRAM 3

```
$IBSYS
$ID R02T       WDD521A, X4189
$*               DEAR ØPERATØR, PLEASE MØUNT THE TAPES,AS DESIGNATED ØN
$*               INSTRUCTIØN CARD,ØN A5 AND A6. MY ØUTPUT UNITS ARE SHØWN
$*               ØN THE INSTRUCTIØN CARD AND,WHEN NUMBER ØF ØUTPUT UNITS
$*               EXCEEDS 5, THE ØN-LINE PRINTER.
```

```
$*                SAVE MY ØUTPUT TAPES.
$*                RESTART MACHINE. THE MACHINE WILL READ AND REWIND A5 AND
$*                A6,SEQUENTIALLY AND CYCLICALLY,UNTIL ALL TAPES ARE READ.
$*                AFTER A5 ØR A6 IS REWØUND,MØUNT,IMMEDIATELY,THE NEXT TAPE
$*                AS SHØWN ØN INSTRUCTIØN CARD.
$PAUSE
$DATE             061065
$EXECUTE          IBJØB
$IBJØB            GØ
C      STATIØN SCHEDULING,SUBPRØGRAM 3 ØF PHASE 1
       LØGICAL EØF
       CØMMØN/SYSEØF/EØF
       EØF=.FALSE.
       CALL SYSGØ(34)
       CALL SYSGØ(35)
       INTEGER REELS,Z,DØTS2,DASH,TNSTAT,TNSATL
       DATA IEND/6HEND ØF/,JWØRD/6HOC     /
      0DIMENSIØN NUM(100,72),NØT(720),MØT(720),JSTAT(100),MSATL(100),NSU(
      110),JFL(72)
       READ 2,DØTS2,DASH
     2 FØRMAT (2A2)
       READ 3,TNSTAT,TNSATL,REELS
     3 FØRMAT (3I3)
       L=1
       DØ 4 NS=1,TNSTAT
       READ 5,JK,KSTAT,NSATL,(NUM(JK,I),NØT(I),I=1,NSATL)
     5 FØRMAT (I3,1X,A6,1X,I2,1X,72(I2,A2))
       IF (JK-NS)6,7,6
     6 PRINT 8
     80FØRMAT (1H0,118HSTØPPING----DATA CARDS ØF STATIØNS ARE DISØRDERED.
      1 CLEAR MACHINE AND PLACE DATA CARDS IN CØRRECT ØRDER. RERUN PRØGRA
      2M.)
       GØTØ 9
     7 JSTAT(NS)=KSTAT
       MSATL(NS)=NSATL
       DØ 10 K=1,NSATL
       MØT(L)=NØT(K)
       L=L+1
    10 CØNTINUE
     4 CØNTINUE
       NS=1
       NU=1
    11 NSPU=1
       MANY=0
    12 MANY=MANY+2*MSATL(NS)+2
       IF (MANY-126)13,14,15
    13 IF (NS-TNSTAT)16,14,16
    16 NSPU=NSPU+1
       NS=NS+1
       GØTØ 12
    15 NSPU=NSPU-1
       NS=NS-1
       IF (NSPU)14,17,14
    17 PRINT 18
    180FØRMAT (1H0,87HSTØPPING----NSPU = 0 . 132 CHARACTER-WIDE PRINT-ØUT
      1 SHEET IS INSUFFICIENT FØR THIS JØB.)
       GØTØ 9
    14 NSU(NU)=NSPU
       IF (NS-TNSTAT)19,20,19
    19 NS=NS+1
```

```
      NU=NU+1
      GOTO 11
   20 JXX=NU
      IF (JXX-6)21,22,23
   23 IF (JXX-7)22,24,25
   25 IF (JXX-8)24,26,27
   27 IF (JXX-9)26,28,29
   29 IF (JXX-10)28,30,31
   22 PRINT 32
   320FORMAT (1H0,67HOPERATOR ACTION PAUSE----DIAL TAPE UNIT A4 TO A7.
     1RESTART MACHINE.)
      PAUSE
      REWIND 7
      GOTO 21
   24 PRINT 33
   330FORMAT (1H0,70HOPERATOR ACTION PAUSE----DIAL A4 TO A7 AND B4 TO B7
     1.  RESTART MACHINE.)
      PAUSE
      REWIND 7
      REWIND 17
      GOTO 21
   26 PRINT 34
   340FORMAT (1H0,80HOPERATOR ACTION PAUSE----DIAL A4 TO A7, B4 TO B7 AN
     1D B2 TO B8.  RESTART MACHINE.)
      PAUSE
      REWIND 7
      REWIND 17
      REWIND 18
      GOTO 21
   28 PRINT 35
   350FORMAT (1H0,90HOPERATOR ACTION PAUSE----DIAL A4 TO A7, B4 TO B7, B
     12 TO B8 AND B1 TO B9.  RESTART MACHINE.)
      PAUSE
      REWIND 7
      REWIND 17
      REWIND 18
      REWIND 19
      GOTO 21
   30 PRINT 36
   360FORMAT (1H0,119HOPERATOR ACTION PAUSE----REPLACE REEL ON A2 WITH A
     1N OUTPUT REEL. DIAL A2 TO A8,A4 TO A7,B4 TO B7,B2 TO B8 AND B1 TO
     2B9./1H0,25X,16HRESTART MACHINE.)
      PAUSE
      REWIND 7
      REWIND 17
      REWIND 18
      REWIND 19
      GOTO 21
   31 PRINT 37
   370FORMAT (1H0,65HSTOPPING----INSUFFICIENT NO. OF PHYSICAL TAPE UNITS
     1 FOR THIS JOB.)
      GOTO 9
   21 READ 38,FDATE,FTIME,LDATE,LTIME
   38 FORMAT (4A6)
      DO 49 J=1,JXX
      GOTO (50,51,52,53,54,55,56,57,58,59),J
   50 WRITE (3,60)FDATE,FTIME,LDATE,LTIME,J,JXX
      GOTO 49
   51 WRITE (15,60)FDATE,FTIME,LDATE,LTIME,J,JXX
      GOTO 49
```

```
52 WRITE (16,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
53 WRITE (21,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
54 WRITE (22,60)FDATE,FTIME,LDATE,LTIMF,J,JXX
   GØTØ 49
55 WRITE (7,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
56 WRITE (17,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
57 WRITE (18,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
58 WRITE (19,60)FDATE,FTIME,LDATE,LTIME,J,JXX
   GØTØ 49
59 WRITE (8,60)FDATE,FTIMF,LDATE,LTIME,J,JXX
600FØRMAT (1H1/1H0/1H0/1H0/1H0/1H0/1H0/1H0/1H0/1H0/1H0/1H0,49X,27HØPE
  1RATIØNAL PREDICTIØNS FØR/1H ,53X,18HSTATIØN SCHEDULING/1H0,49X,4HF
  2RØM,2X,A6,1X,A6/1H ,51X,2HTØ,2X,A6,1X,A6/1H0,56X,5H(PART,1X,I2,1X,
  32HØF,1X,I2,1H)/1H1/1H0,77HID.  ID.NØ.  SATELLITE  START-DATE  STAR
  4T-TIME  STØP-DATE  STØP-TIME  MIN.EL./1H ,24X,50HY M D       H M S
  5       Y M D       H M S       DEG.)
49 CØNTINUF
   JRE=1
   JJ2=1
   DØ 61 JT=1,TNSATL
   READ 62,Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
62 FØRMAT (A1,1X,A5,1X,A6,A4,1X,A6,1X,A6,1X,A6,1X,A6,1X,A5)
   DØ 63 J=1,JXX
   GØTØ (64,65,66,67,68,69,70,71,72,73),J
64 WRITE (3,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
65 WRITE (15,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
66 WRITE (16,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
67 WRITE (21,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
68 WRITE (22,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
69 WRITE (7,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
70 WRITE (17,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
71 WRITE (18,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
72 WRITE (19,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
   GØTØ 63
73 WRITE (8,74)Z,IDENT,JSATL,NØS,JSDAT,JSTIM,JXDAT,JXTIM,JPSI
74 FØRMAT (1H ,A1,4X,A5,3X,A6,A4,1X,A6,6X,A6,6X,A6,5X,A6,5X,A5)
63 CØNTINUE
61 CØNTINUF
   NU=1
   NS=0
   L=0
   K=0
75 NX=1
81 NS=NS+1
   JT=1
76 L=L+1
   K=K+1
```

```
    NOT(L)=MOT(K)
    IF (JT-MSATL(NS))77,78,77
 77 JT=JT+1
    GOTO 76
 78 IF (NX-NSU(NU))79,80,79
 79 NX=NX+1
    L=L+1
    NOT(L)=DOTS2
    GOTO 81
 80 IF (NU-JXX)82,83,82
 82 NU=NU+1
    GOTO 75
 83 READ (5,84)KWORD,KDATE,KTIM,JK,I,KDAY,KEL
 84 FORMAT (A2,1X,I6,1X,A4,1X,I3,1X,I3,1X,I3,2X,A2)
    IF (KWORD-JWORD)83,85,83
 85 JDATE=KDATE
191 LXX=0
    LSU=0
    NX=0
    DO 86 J=1,JXX
    MXX=NSU(J)
    LSU=LSU+MXX
    DO 87 MN=1,MXX
    NX=NX+1
    LXX=LXX+MSATL(NX)
 87 CONTINUE
    LXX=LXX+MXX-1
    GOTO (88,89,90,91,92,93,94,95,96,97),J
 88 WRITE (3,98)KDATE,KDAY,(JSTAT(NS),NS=1,LSU),(NOT(L),L=1,LXX)
    GOTO 108
 89 WRITE (15,99)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 90 WRITE (16,100)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 91 WRITE (21,101)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 92 WRITE (22,102)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 93 WRITE (7,103)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 94 WRITE (17,104)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 95 WRITE (18,105)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 96 WRITE (19,106)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
    GOTO 108
 97 WRITE (8,107)KDATE,KDAY,(JSTAT(NS),NS=N1,LSU),(NOT(L),L=N2,LXX)
108 N1=LSU+1
    N2=LXX+1
 86 CONTINUE
    JSKIP=1
109 JTIM=KTIM
    MJK=1
    NX=1
    J=1
168 N=1
    NS=1
110 K=JK
    M=1
111 IF (MJK-K)112,113,112
```

```
112 NSATL=MSATL(NX)
    DO 114 M=1,NSATL
    JEL(N)=DASH
    N=N+1
114 CONTINUE
    NX=NX+1
    IF (NS-NSU(J))115,116,115
115 NS=NS+1
    JEL(N)=DOTS2
    N=N+1
    GOTO 129
116 N=N-1
    GOTO (117,118,119,120,121,122,123,124,125,126),J
117 WRITE (3,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
118 WRITE (15,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
119 WRITE (16,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
120 WRITE (21,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
121 WRITE (22,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
122 WRITE (7,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
123 WRITE (17,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
124 WRITE (18,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
125 WRITE (19,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 128
126 WRITE (8,127)JTIM,(JEL(IN),IN=1,N)
127 FORMAT (1H ,A4,3X,72A2)
128 J=J+1
    N=1
    NS=1
129 MJK=MJK+1
    M=1
    GOTO 111
113 IF (NUM(K,M)-I)130,131,130
130 JEL(N)=DASH
132 M=M+1
    N=N+1
    GOTO 113
131 JEL(N)=KEL
133 GOTO (134,135),JJ2
134 READ (5,136)KDATE,KTIM,JK,I,KDAY,KEL
    IF (EOF)GOTO 137
    GOTO 138
135 READ (6,136)KDATE,KTIM,JK,I,KDAY,KEL
    IF (EOF)GOTO 137
    GOTO 138
136 FORMAT (3X,I6,1X,A4,1X,I3,1X,I3,1X,I3,2X,A2)
137 EOF=.FALSE.
    IF (JRE-REELS)139,140,139
139 JRE=JRE+1
    GOTO (141,142),JJ2
141 REWIND 5
    JJ2=2
    GOTO 135
```

```
142 REWIND 6
    JJ2=1
    GOTO 134
140 JRE=IEND
    GOTO (143,144),JJ2
143 REWIND 5
    GOTO 145
144 REWIND 6
    GOTO 145
138 IF (JDATE-KDATE)145,146,145
146 IF (JTIM-KTIM)145,147,145
147 IF (K-JK)148,149,148
149 IF (NUM(K,M)-I)132,133,150
150 PRINT 151
151 FORMAT (1H0,36HSTOPPING----SORT ERROR ON INPUT TAPE)
    GOTO 9
148 IF (M-MSATL(NX))152,153,152
152 M=M+1
    N=N+1
    JEL(N)=DASH
    GOTO 148
153 NX=NX+1
    IF (NS-NSU(J))154,155,154
154 N=N+1
    NS=NS+1
    JEL(N)=DOTS2
156 N=N+1
    MJK=MJK+1
    GOTO 110
155 GOTO (157,158,159,160,161,162,163,164,165,166),J
157 WRITE (3,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
158 WRITE (15,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
159 WRITE (16,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
160 WRITE (21,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
161 WRITE (22,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
162 WRITE (7,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
163 WRITE (17,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
164 WRITE (18,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
165 WRITE (19,127)JTIM,(JEL(IN),IN=1,N)
    GOTO 167
166 WRITE (8,127)JTIM,(JEL(IN),IN=1,N)
167 J=J+1
    MJK=MJK+1
    GOTO 168
145 IF (M-MSATL(NX))169,170,169
169 M=M+1
    N=N+1
    JEL(N)=DASH
    GOTO 145
170 NX=NX+1
    IF (NS-NSU(J))171,172,171
171 N=N+1
    NS=NS+1
```

```
      JEL(N)=DOTS2
      M=0
      GOTO 169
  172 GOTO (173,174,175,176,177,178,179,180,181,182),J
  173 WRITE (3,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  174 WRITE (15,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  175 WRITE (16,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  176 WRITE (21,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  177 WRITE (22,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  178 WRITE (7,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  179 WRITE (17,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  180 WRITE (18,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  181 WRITE (19,127)JTIM,(JEL(IN),IN=1,N)
      GOTO 183
  182 WRITE (8,127)JTIM,(JEL(IN),IN=1,N)
  183 IF (J-JXX)184,185,184
  184 J=J+1
      N=0
      NS=1
      M=0
      GOTO 169
  185 IF (JRF-IFND)186,187,186
  186 IF (JDATE-KDATE)188,189,188
  188 DO 192 J=1,JXX
      GOTO (193,194,195,196,197,198,199,200,201,202),J
  193 WRITE (3,203)
      GOTO 192
  194 WRITE (15,203)
      GOTO 192
  195 WRITE (16,203)
      GOTO 192
  196 WRITE (21,203)
      GOTO 192
  197 WRITE (22,203)
      GOTO 192
  198 WRITE (7,203)
      GOTO 192
  199 WRITE (17,203)
      GOTO 192
  200 WRITE (18,203)
      GOTO 192
  201 WRITE (19,203)
      GOTO 192
  202 WRITE (8,203)
  203 FORMAT (1H ,3HEND)
  192 CONTINUE
      GOTO 85
  189 IF (JSKIP-57)190,191,190
  190 JSKIP=JSKIP+1
      GOTO 109
  187 DO 204 J=1,JXX
      GOTO (205,206,207,208,209,210,211,212,213,214),J
```

```
 205 WRITF (3,215)JRE
     GOTO 204
 206 WRITE (15,215)JRE
     GOTO 204
 207 WRITE (16,215)JRE
     GOTO 204
 208 WRITF (21,215)JRE
     GOTO 204
 209 WRITF (22,215)JRE
     GOTO 204
 210 WRITF (7,215)JRE
     GOTO 204
 211 WRITF (17,215)JRE
     GOTO 204
 212 WRITF (18,215)JRE
     GOTO 204
 213 WRITE (19,215)JRE
     GOTO 204
 214 WRITE (8,215)JRE
 215 FORMAT (1H ,3HFND/1H ,A6,4H RUN)
 204 CONTINUF

  98 FORMAT
(FORMAT MUST BE UPDATED FOR FACH RUN. SEE APPENDIX B)

  99 FORMAT
(FORMAT MUST BF UPDATED FOR EACH RUN. SEE APPENDIX B)

 100 FORMAT
(FORMAT MUST BE UPDATED FOR FACH RUN. SEE APPENDIX B)

 101 FORMAT
(FORMAT MUST BF UPDATED FOR FACH RUN. SEE APPFNDIX B)

 102 FORMAT
(FORMAT MUST BE UPDATED FOR EACH RUN. SEE APPFNDIX B)

 103 FORMAT
(FORMAT MUST BE UPDATED FOR EACH RUN. SEE APPENDIX B)

 104 FORMAT
(FORMAT MUST BE UPDATED FOR EACH RUN. SEE APPENDIX B)

 105 FORMAT
(FORMAT MUST BE UPDATED FOR FACH RUN. SEE APPFNDIX B)

 106 FORMAT
(FORMAT MUST BF UPDATED FOR FACH RUN. SEE APPFNDIX B)

 107 FORMAT
(FORMAT MUST BE UPDATED FOR EACH RUN. SEE APPFNDIX B)

   9 STOP
     FND
$IBMAP UN15
       ENTRY    .UN15.
.UN15. PZF      UNIT15
UNIT15 FILE     ,B(1),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       FND
```

```
$IBMAP UN16
       ENTRY    .UN16.
.UN16. PZE      UNIT16
UNIT16 FILE     ,B(2),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN21
       ENTRY    .UN21.
.UN21. PZE      UNIT21
UNIT21 FILE     ,C(1),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN22
       ENTRY    .UN22.
.UN22. PZE      UNIT22
UNIT22 FILE     ,C(2),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN07
       ENTRY    .UN07.
.UN07. PZE      UNIT07
UNIT07 FILE     ,A(3),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN17
       ENTRY    .UN17.
.UN17. PZE      UNIT17
UNIT17 FILE     ,B(3),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN18
       ENTRY    .UN18.
.UN18. PZE      UNIT18
UNIT18 FILE     ,B(4),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN19
       ENTRY    .UN19.
.UN19. PZE      UNIT19
UNIT19 FILE     ,B(5),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END
$IBMAP UN08
       ENTRY    .UN08.
.UN08. PZE      UNIT08
UNIT08 FILE     ,A(4),READY,INOUT,BLK=22,MULTIREEL,BCD,HOLD,NOLIST
       END

$IBLDR FRWB
(RELOCATABLE BINARY DECK FOR SUBROUTINE FRWB)

$IBLDR FRWD
(RELOCATABLE BINARY DECK FOR SUBROUTINE FRWD)

$IBLDR FSLI
(RELOCATABLE BINARY DECK FOR SUBROUTINE FSLI)

$IBLDR FXEM
(RELOCATABLE BINARY DECK FOR SUBROUTINE FXEM)

$IBLDR NOSYS
(RELOCATABLE BINARY DECK FOR SUBROUTINE NOSYS)

$IBLDR GOSYS
(RELOCATABLE BINARY DECK FOR SUBROUTINE GOSYS)

(END-OF-FILE CARD)
$FMSYS
(END-OF-FILE CARD)
```

APPENDIX B

FORMULATED FORMATS OF SUBPROGRAM 3

```
  980FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(1)-4)X,A6,(2*MSA
    1TL(2)-4)X,A6,------,A6,(2*MSATL(NSU(1))-4)X,A6/1H ,3HH M,4X,72A2)

  990FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(1)+1)-4)X,A6
    1,(2*MSATL(NSU(1)+2)-4)X,A6,------,A6,(2*MSATL(NSU(2))-4)X,A6/1H ,3
    2HH M,4X72A2)

 1000FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(2)+1)-4)X,A6
    1,(2*MSATL(NSU(2)+2)-4)X,A6,------,A6,(2*MSATL(NSU(3))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1010FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIMF,3X,A6,(2*MSATL(NSU(3)+1)-4)X,A6
    1,(2*MSATL(NSU(3)+2)-4)X,A6,------,A6,(2*MSATL(NSU(4))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1020FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(4)+1)-4)X,A6
    1,(2*MSATL(NSU(4)+2)-4)X,A6,------,A6,(2*MSATL(NSU(5))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1030FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(5)+1)-4)X,A6
    1,(2*MSATL(NSU(5)+2)-4)X,A6,------,A6,(2*MSATL(NSU(6))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1040FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(6)+1)-4)X,A6
    1,(2*MSATL(NSU(6)+2)-4)X,A6,------,A6,(2*MSATL(NSU(7))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1050FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(7)+1)-4)X,A6
    1,(2*MSATL(NSU(7)+2)-4)X,A6,------,A6,(2*MSATL(NSU(8))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1060FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(8)+1)-4)X,A6
    1,(2*MSATL(NSU(8)+2)-4)X,A6,------,A6,(2*MSATL(NSU(9))-4)X,A6/1H ,3
    2HH M,4X,72A2)

 1070FORMAT (1H1,7X,I6,2X,I3/1H ,4HTIME,3X,A6,(2*MSATL(NSU(9)+1)-4)X,A6
    1,(2*MSATL(NSU(9)+2)-4)X,A6,------,A6,(2*MSATL(NSU(10))-4)X,A6/1H ,
    23HH M,4X,72A2)
```